Choice Models in Operations

Lecture 8 : Mixed logit models

Instructor: Srikanth Jagabathula *Scribe: Omar El Housni*

We want to capture the heterogeneity in the preferences of customers through their " taste" vectors. This results in a mixture of MNL models which represents a very general and flexible model class. Depending on the form of the mixture distribution, there are broadly two types of ML models:

- **LC-MNL (Latent Class-MNL):** Discrete distribution with a finite support: K classes.
- **RPL (Random Parameters Logit):** Continuous mixture distribution, (it's also called random coefficients MNL model).

1 RPL model class

Consider $f(.)$ a distribution over the parameters of the logit model. The choice probabilities are given by

$$\mathbb{P}(j|S) = \int_{\underline{\beta} \in \mathbb{R}^d} \frac{e^{\underline{\beta}^T x_j}}{\sum_{i \in S_t} e^{\underline{\beta}^T x_i}} f(\underline{\beta}) d\underline{\beta}$$

where the products are represented by d-dimensional feature vector.

The most popular mixing distribution is the Multivariate Gaussian distribution with mean $\underline{\mu} \in \mathbb{R}^{d \times 1}$ and variance-covariance matrix $\Sigma_{d \times d} \succeq 0$. The simplest member of this family is obtained by $\Sigma = \sigma^2 I_{d \times d}$ where $I_{d \times d}$ is the $d \times d$ identity matrix. In this case, we have

$$\mathbb{P}(j|S) = \int_{\underline{\beta} \in \mathbb{R}^d} \frac{e^{\underline{\beta}^T x_j}}{\sum_{i \in S_t} e^{\underline{\beta}^T x_i}} \prod_{k=1}^{d} f(\beta_k) d\beta_k.$$

Remarks:

1. **Modeling / Incorporating individual-level features:** Suppose we have choice observations from customers who are represented as K-dimensional feature vector $\underline{z} \in \mathbb{R}^k$. For example, let say $k = 2$ and the features are age and income. A customer for instance who is 24 years old and has an annual income of 100k is represented by a two dimensional vector $[24 \;\; 100k]$.

   To capture individual-level features, we consider the following hierarchical model:

   (a) Sample $\underline{\beta}$ according to $\underline{\beta} = A\underline{z} + \underline{\epsilon}$ where $\underline{\epsilon}$ is random.

   (b) Choose items according to an MNL model with taste $\underline{\beta}$.

2. **Generality of the ML model** At the aggregate level, the ML model can capture the MNL and NL models.

   (a) MNL $\in$ ML : Put a point mass on one $\underline{\beta}$ vector.

(b) NL $\in$ ML:
Recall the NL model

$$U_{lj} = r^T \underline{x}_{lj} + \tilde{\epsilon}_l + \tilde{\epsilon}_{lj}$$

We want to have

$$U_{lj} = \underline{\beta}^T \underline{y}_{lj} + \tilde{\epsilon}_{lj}$$

We can take

$$\underline{y}_{lj} = [\underline{x}_{lj} \,|e_l]$$

and

$$\underline{\beta} = [\underline{r} \mid \tilde{\epsilon}_1 \mid \tilde{\epsilon}_2 \mid \ldots |\tilde{\epsilon}_n]$$

where e_l is the unit vector in $\mathbb{R}^n$ with 1 at position l and 0 otherwise.

(c) Most generally, McFadden and Train (2000) showed that the ML model can approximate any RUM model "arbitrarely closely".

Proof Idea: Consider RUM model of the form $U_j = \underline{\alpha}^T \underline{x}_j$ where $\underline{\alpha} \overset{D}{\sim} F(.)$. From this we can construct the following instance of the ML class:

$$\tilde{U}_j = \underline{\alpha}^T \underline{x}_j + \frac{1}{c}\epsilon_j$$

where $\epsilon_j \overset{D}{\sim}$ Gumbel$(0,1)$ and $\underline{\alpha} \overset{D}{\sim} F(.)$

3. **Comment on how these models are used in Econ & Marketing:** Generally speaking, customer-level heterogeneity is considered a " nuisance" in Econ whereas it is the 'focus' in Marketing. In Econ, we focus more on understanding the aggregate preferences of a population in order to derive policy implications. In particular, the precise form of distribution is less relevant and the focus is more on the mean/var of the parameters. In marketing, the focus typically is in individual-level heterogeneity so that the firm can customize its marketing activity to individual preferences. In particular, we focus on recovering the form of the distribution.

2 Taking the models to data

We focus now on estimating the parameters of an MNL model.

Data: We have choice observations of the form $(j_t, S_t),\ t = 1, \ldots T$ where product j_t was chosen from S_t in choice instance t. The data are generated according to an MNL model where

$$U_j = \beta_j + \epsilon_j$$

for $j = 1, \ldots, n$ and ϵ_j are iid Gumbel(0,1).

Goal: Estimate the parameters β_j of the model. We carry out the estimation using the max likelihood estimation (MLE) technique. We first write down the likelihood of the data

$$L(\underline{\beta}) = \prod_{t=1}^{T} \frac{e^{\beta_{j_t}}}{\sum_{i \in S_t} e^{\beta_i}}.$$

We now take the log to get the data log-likelihood function

$$l(\underline{\beta}) = \log L(\underline{\beta}) = \sum_{t=1}^{T} \log \frac{e^{\beta_{j_t}}}{\sum_{i \in S_t} e^{\beta_i}} = \sum_{t=1}^{T} \left(\beta_{j_t} - \log \sum_{i \in S_t} e^{\beta_i} \right).$$

The above expression can be simplified further. Suppose the data consist of L different offer sets $S_1, S_2, \ldots S_L$. We define the following counts:

$$\begin{aligned} c_{jr} &= \#\text{ of times j was purchased when } S_r \text{ was on offer} \\ c_{j.} &= \sum_{r=1}^{L} c_{jr} \text{ is the number of times j was choosen in the data} \\ c_{.r} &= \sum_{j \in S_r} c_{jr} \text{ is the number of times } S_r \text{ was offered in the data} \end{aligned}$$

Under this notations, we can write

$$\begin{aligned} l(\underline{\beta}) &= \sum_{t=1}^{T} \beta_{j_t} - \sum_{t=1}^{T} \log \sum_{i \in S_t} e^{\beta_i} \\ &= \sum_{j=1}^{n} c_{j.} \beta_j - \sum_{r=1}^{L} c_{.r} \log \sum_{i \in S_r} e^{\beta_i} \end{aligned}$$

Therefore $c_{j.}$ and $c_{.r}$ are the sufficient data for the purpose of estimating MNL parameters. We want to solve the following maximization problem

$$\max_{\underline{\beta}} \; l(\underline{\beta}) = \max_{\underline{\beta}} \; \sum_{j=1}^{n} c_{j.} \beta_j - \sum_{r=1}^{L} c_{.r} \log \sum_{i \in S_r} e^{\beta_i}$$

We first focus on the FOC, consider

$$\begin{aligned} \frac{\partial l(\underline{\beta})}{\partial \beta_k} &= c_{k.} - \sum_{r: k \in S_r} c_{.r} \frac{e^{\beta_k}}{\sum_{i \in S_r} e^{\beta_i}} \\ &= c_{k.} - \sum_{r: k \in S_r} c_{.r} q_{kr}(\underline{\beta}) \\ &= c_{k.} - \sum_{r=1}^{L} c_{.r} q_{kr}(\underline{\beta}) \end{aligned}$$

where $q_{kr}(\underline{\beta})$ is the probability of choosing k from S_r under parameter vector $\underline{\beta}$ and $q_{kr}(\underline{\beta}) = 0$ if $k \notin S_r$.

Setting the partial derivatives equal to 0 we get

$$c_{k.} = \sum_{r=1}^{L} c_{.r} q_{kr}(\underline{\beta}) \qquad k = 1, \ldots, n$$

The above set of equations say that at the stationary point, the observed number of purchases of each item in the data must be equal to the expected number of purchases for the product. To show that the optimal solution occurs at a stationary point, we compute the Hessian and show that it is negative-semi definite. Let us compute the Hessian.

$$\begin{aligned}\frac{\partial^2 l(\underline{\beta})}{\partial\beta_k\beta_{k'}} &= -\sum_{r=1}^{L} c_{\cdot r}\frac{\partial}{\partial\beta_{k'}}q_{kr}(\underline{\beta}) \\ &= -\sum_{r=1}^{L} c_{\cdot r}\frac{\partial}{\partial\beta_{k'}}\frac{e^{\beta_k}}{\sum_{i\in S_r}e^{\beta_i}}\end{aligned}$$

We have

$$\frac{\partial}{\partial\beta_{k'}}\frac{e^{\beta_k}}{\sum_{i\in S_r}e^{\beta_i}} = \begin{cases} -\frac{e^{\beta_k}e^{\beta_{k'}}}{\left(\sum_{i\in S_r}e^{\beta_i}\right)^2} & \text{if } k\neq k', k'\in S_r \\ \frac{e^{\beta_k}}{\sum_{i\in S_r}e^{\beta_i}} - \left(\frac{e^{\beta_k}}{\sum_{i\in S_r}e^{\beta_i}}\right)^2 & \text{if } k = k' \end{cases}$$

With our notations, we have

$$\frac{\partial}{\partial\beta_{k'}}q_{kr}(\underline{\beta}) = \begin{cases} -q_{kr}(\underline{\beta})q_{k'r}(\underline{\beta}) & \text{if } k\neq k', k'\in S_r \\ q_{kr}(\underline{\beta})(1-q_{kr}(\underline{\beta})) & \text{if } k = k' \end{cases}$$

Hence,

$$\frac{\partial^2 l(\underline{\beta})}{\partial\beta_k\beta_{k'}} = \begin{cases} \sum_{r=1}^{L} c_{\cdot r}q_{kr}(\underline{\beta})q_{k'r}(\underline{\beta}) & \text{if } k\neq k', k'\in S_r \\ -\sum_{r=1}^{L} c_{\cdot r}q_{kr}(\underline{\beta})(1-q_{kr}(\underline{\beta})) & \text{if } k = k' \end{cases}$$

We want to show that the Hessian is semi-definite positive. First let us introduce the following definition and Theorem.

Definition 1 *A matrix D is diagonally dominant if $|D_{ii}| \geq \sum_{j\neq i}|D_{ij}|\ \forall i$.*

Theorem 1 *If D is a symmetric diagonally dominant matrix with non-negative diagonal elements, then D is positive semi-definite.*

Let us show that H is diagonally dominant

$$\begin{aligned}\sum_{k'\neq k}\left|\frac{\partial^2 l(\underline{\beta})}{\partial\beta_k\beta_{k'}}\right| &= \sum_{r=1}^{L} c_{\cdot r}q_{kr}(\underline{\beta})\sum_{k'\neq k}q_{k'r}(\underline{\beta}) \\ &= \sum_{r=1}^{L} c_{\cdot r}q_{kr}(\underline{\beta})(1-q_{kr}(\underline{\beta})) \\ &= \left|\frac{\partial^2 l(\underline{\beta})}{\partial^2\beta_k}\right|\end{aligned}$$

where the last equality follows from the fact that $\sum_{k'} q_{k'r}(\underline{\beta}) = 1 \ \forall r$. Hence $-H$ is diagonally dominant and has non-negative diagonal elements. Therefore, from the above Theorem $-H$ is positive semi-definite and consequently H is negative semi-definite as desired.

Newton's method for finding the optimal solution

Newton's method is designed to find the root of a function. In our case, our objective is to find the roots of the gradient of the log-likelihood function. In particular $F(\underline{\beta}) = \nabla l(\underline{\beta})$. Since Newton's method is an iterative method, we focus on the update step in each iteration i.e. when given the current solution, $\underline{\beta}^{(t)}$ we want to find a " better" solution $\underline{\beta}^{(t+1)}$.

The Newton's update step is

$$\underline{\beta}^{(t+1)} = \underline{\beta}^{(t)} - \left(\nabla F(\underline{\beta}^{(t)}\right)^{-1} F(\underline{\beta}^{(t)}).$$

In our setting, we get

$$\underline{\beta}^{(t+1)} = \underline{\beta}^{(t)} - \left(\nabla^2 l(\underline{\beta}^{(t)})\right)^{-1} \nabla l(\underline{\beta}^{(t)}). \tag{1}$$

We are now ready to state the algorithm.

Algorithm

1. Start with an initial estimate $\underline{\beta}^{(0)}$.
2. In each iteration, update $\underline{\beta}^{(t)}$ according to (1).
3. Repeat until a stopping condition is met. There are two commonly used stopping conditions:
   (a) Stop while $\|\nabla l(\underline{\beta}^{(t)})\|_2 < \epsilon$ where ϵ is a tolerance parameter.
   (b) Stop while $\|\underline{\beta}^{(t+1)} - \underline{\beta}^{(t)}\|_2 < \epsilon$.